Google

Web Images Groups News Froogle Local more »

vmm extensible firmware Search Advanced Search Preferences

O Search the Web ◉ Search English and German pages

Web Results **1 - 10** of about **321 English and German** pages for **vmm extensible firmware**. (**0.39** seconds)

memory management: Information From Answers.com
µCLinux, PMMU (technology). chunking (computing), Harlequin Inc. Accent Network Operating System Kernel, **Extensible Firmware** Interface ...
www.answers.com/topic/memory-management - 27k - Cached - Similar pages

HP OpenVMS systems documentation
EFI: **Extensible firmware** interface. EFI's purpose is to initialize the hardware and start booting an operating system. For more information, see Intel's EFI ...
h71000.www7.hp.com/doc/82final/6673/6673pro_010.html - 43k - Cached - Similar pages

Computer Acronyms, Definitions, Slang, and Lingo
EFI, **Extensible Firmware** Interface. EFT, Electronic Funds Transfer ... **VMM**, Virtual Memory Manager. VMS, Virtual Machine System ...
www.web-friend.com/help/lingo/pcacnm.html - 211k - Cached - Similar pages

AXCEL216 / MDGx GLOSSARY
EFI = **Extensible Firmware** Interface future BIOS standard (Intel). EGA = Extended Graphics Adapter. ... **VMM** (1) = Virtual Memory Manager (Microsoft Windows). ...
www.mdgx.com/glossary.htm - 86k - Cached - Similar pages

[PDF] Portierung von Fiasco auf IA-64
File Format: PDF/Adobe Acrobat - View as HTML
if an **Extensible Firmware** Interface (EFI) implementation is provided ...
EFI **Extensible Firmware** Interface, is a well defined software interface for machine ...
os.inf.tu-dresden.de/papers_ps/warg-beleg.pdf - Similar pages

Sysinternals Freeware - Information for Windows NT and Windows ...
These functions are for managing IA64 **Extensible Firmware** Interface device drivers. On non-IA64 systems these return STATUS_NOT_IMPLEMTNED. ...
www.sysinternals.com/ntw2k/info/ntdll.shtml - 197k - Cached - Similar pages

[PDF] The Integration of Virtual Memory Management and Interprocess ...
File Format: PDF/Adobe Acrobat
ming]: Microcode Applications-**firmware** ... designed to be transparently **extensible** by user-state processes. This permits ...
portal.acm.org/ft_gateway.cfm?id=214422&type=pdf - Similar pages

[PDF] Intel Itanium Processor Family System Abstraction Layer Specification
File Format: PDF/Adobe Acrobat
A companion document, the **Extensible Firmware** Interface Specification, ... described in later sections of this document and the **Extensible Firmware** ...
developer.intel.com/design/ itanium/downloads/24535907.pdf - Similar pages

[PDF] THE DESIGN AND APPLICATION OF AN **EXTENSIBLE** OPERATING SYSTEM ...
File Format: PDF/Adobe Acrobat - View as HTML
Central to this thesis is our own **extensible** operating system, ... of all protection mechanisms in a computing system, including hardware, **firmware**, ...

www.ece.cmu.edu/~leendert/publications/Thesis.pdf - Similar pages

[PDF] THE DESIGN AND APPLICATION OF AN **EXTENSIBLE** OPERATING SYSTEM ...
File Format: PDF/Adobe Acrobat - View as HTML
In this thesis we present a simple **extensible** system for building ... of all protection mechanisms in a computing system, including hardware, **firmware**, ...
www.cs.vu.nl/res/theses/doorn_thesis.pdf - Similar pages

Gooooooooooogle ▶
Result Page: 1 2 3 4 5 6 7 8 9 10 **Next**

Free! Google Desktop Search: Search your own computer. Download now.
Find: emails - files - chats - web history - media - PDF

vmm extensible firmware Search

Search within results | Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

©2005 Google

PORTAL USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

vmm efi extensible firmware

Search

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **vmm efi extensible firmware** Found **785** of **156,259**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Devirtualizable virtual machines enabling general, single-node, online maintenance

David E. Lowell, Yasushi Saito, Eileen J. Samberg

October 2004 **Proceedings of the 11th international conference on Architectural support for programming languages and operating systems**, Volume 32 , 38 , 39 Issue 5 , 5 , 11

Full text available: pdf(174.01 KB) Additional Information: full citation, abstract, references, index terms

Maintenance is the dominant source of downtime at high availability sites. Unfortunately, the dominant mechanism for reducing this downtime, cluster rolling upgrade, has two shortcomings that have prevented its broad acceptance. First, cluster-style maintenance over many nodes is typically performed a few nodes at a time, mak-ing maintenance slow and often impractical. Second, cluster-style maintenance does not work on single-node systems, despite the fact that their unavailability during mainte ...

Keywords: availability, online maintenance, planned downtime, virtual machines

2 Virtual machine monitors: Terra: a virtual machine-based platform for trusted computing

Tal Garfinkel, Ben Pfaff, Jim Chow, Mendel Rosenblum, Dan Boneh

October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**

Full text available: pdf(140.31 KB) Additional Information: full citation, abstract, references, citings, index terms

We present a flexible architecture for trusted computing, called Terra, that allows applications with a wide range of security requirements to run simultaneously on commodity hardware. Applications on Terra enjoy the semantics of running on a separate, dedicated, tamper-resistant hardware platform, while retaining the ability to run side-by-side with normal applications on a general-purpose computing platform. Terra achieves this synthesis by use of a *trusted virtual machine monitor* (TVMM ...

Keywords: VMM, attestation, authentication, trusted computing, virtual machine, virtual machine monitor

3 An implementation scheme for a virtual machine monitor to be realized on user - microprogrammable minicomputers

B. D. Shriver, J. W. Anderson, L. J. Waguespack, D. M. Hyams, R. A. Bombet
October 1976 **Proceedings of the annual conference**

Full text available: pdf(654.60 KB) Additional Information: full citation, abstract, references, citings, index terms

A virtual machine monitor allows several different operating systems to run concurrently on the same machine. This paper presents the description of a virtual machine monitor and its support structure which can be implemented on a microprogrammable minicomputer or a distributed network of such machines. In our approach, all storage, transformational, input, and output resources of the system are accessed through a mapping mechanism. The design and implementation methodology for an actual re ...

4 Helper threads via virtual multithreading on an experimental itanium® 2 processor-based platform

Perry H. Wang, Jamison D. Collins, Hong Wang, Dongkeun Kim, Bill Greene, Kai-Ming Chan, Aamir B. Yunus, Terry Sych, Stephen F. Moore, John P. Shen
October 2004 **Proceedings of the 11th international conference on Architectural support for programming languages and operating systems**, Volume 39 , 38 , 32 Issue 11 , 5 , 5

Full text available: pdf(225.47 KB) Additional Information: full citation, abstract, references, index terms

Helper threading is a technology to accelerate a program by exploiting a processor's multithreading capability to run ``assist'' threads. Previous experiments on hyper-threaded processors have demonstrated significant speedups by using helper threads to prefetch hard-to-predict delinquent data accesses. In order to apply this technique to processors that do not have built-in hardware support for multithreading, we introduce virtual multithreading (VMT), a novel form of switch-on-event user-level ...

Keywords: DB2 database, PAL, cache miss prefetching, helper thread, itanium processor, multithreading, switch-on-event

5 EASY—an operating system for the QM-1

Charles W. Flink
September 1977 **ACM SIGMICRO Newsletter , Proceedings of the 10th annual workshop on Microprogramming**, Volume 8 Issue 3

Full text available: pdf(733.19 KB) Additional Information: full citation, abstract, references, citings, index terms

The Emulation Aid SYstem is a virtual machine monitor for the Nanodata QM-1 microprogrammable computer. The system is designed to provide the user with an interactive interface for the development and subsequent use of emulations on the QM-1. EASY provides integrated support for: 1) interactive control of multiple, concurrently resident, virtual computers implemented via emulation, 2) input/output from emulations (virtual I/O) to the various real peripherals of the QM-1, and 3) diagnostic d ...

Keywords: Emulation, Intermediate language machines, Microprogramming, Nanodata QM-1, Software engineering, Virtual machine monitors, Virtual machines

6 Kernel korner: About LinuxBIOS

Eric Biederman
December 2001 **Linux Journal**, Volume 2001 Issue 92

Full text available: html(16.22 KB) Additional Information: full citation, index terms

7 Architecture of virtual machines
R. P. Goldberg
March 1973 **Proceedings of the workshop on virtual computer systems**

Full text available: pdf(1.29 MB) Additional Information: full citation, abstract, references, citings, index terms

In this paper we develop a model which represents the addressing of resources by processes executing on a virtual machine. The model distinguishes two maps: the ø-map which represents the map visible to the operating system software running on the virtual machine, and the f-map which is invisible to that software but which is manipulated by the virtual machine monitor running on the real machine. The ø-map maps process names into resource names and the f-map maps virtual resou ...

8 Cellular disco: resource management using virtual clusters on shared-memory multiprocessors
Kinshuk Govil, Dan Teodosiu, Yongqiang Huang, Mendel Rosenblum
August 2000 **ACM Transactions on Computer Systems (TOCS)**, Volume 18 Issue 3

Full text available: pdf(287.05 KB) Additional Information: full citation, abstract, references, citings, index terms, review

Despite the fact that large-scale shared-memory multiprocessors have been commercially available for several years, system software that fully utilizes all their features is still not available, mostly due to the complexity and cost of making the required changes to the operating system. A recently proposed approach, called Disco, substantially reduces this development cost by using a virtual machine monitor that laverages the existing operating system technology. In this paper we present a ...

Keywords: fault containment, resource managment, scalable multiprocessors, virtual machines

9 Cellular Disco: resource management using virtual clusters on shared-memory multiprocessors
Kinshuk Govil, Dan Teodosiu, Yongqiang Huang, Mendel Rosenblum
December 1999 **ACM SIGOPS Operating Systems Review , Proceedings of the seventeenth ACM symposium on Operating systems principles**, Volume 33 Issue 5

Full text available: pdf(1.93 MB) Additional Information: full citation, abstract, references, citings, index terms

Despite the fact that large-scale shared-memory multiprocessors have been commercially available for several years, system software that fully utilizes all their features is still not available, mostly due to the complexity and cost of making the required changes to the operating system. A recently proposed approach, called Disco, substantially reduces this development cost by using a virtual machine monitor that leverages the existing operating system technology.In this paper we present a syste ...

10 Extensible file systems in spring
Yousef A. Khalidi, Michael N. Nelson

December 1993 **ACM SIGOPS Operating Systems Review , Proceedings of the fourteenth ACM symposium on Operating systems principles,** Volume 27 Issue 5

Full text available: pdf(1.47 MB) Additional Information: full citation, abstract, references, citings, index terms

In this paper we describe an architecture for extensible file systems. The architecture enables the extension of file system functionality by composing (or stacking) new file systems on top of existing file systems. A file system that is stacked on top of an existing file system can access the existing file system's files via a well-defined naming interface and can share the same underlying file data in a coherent manner. We describe extending file systems in the context of the Spring operating ...

11 The DCM: a hardware extensible architecture

Robert Edward Boring

October 1985 **Proceedings of the 1985 ACM annual conference on The range of computing : mid-80's perspective: mid-80's perspective**

Full text available: pdf(685.97 KB) Additional Information: full citation, references, index terms

12 Robustness: Using model checking to debug device firmware

Sanjeev Kumar, Kai Li

December 2002 **ACM SIGOPS Operating Systems Review**, Volume 36 Issue SI

Full text available: pdf(1.72 MB) Additional Information: full citation, abstract, references

Device firmware is a piece of concurrent software that achieves high performance at the cost of software complexity. They contain subtle race conditions that make them difficult to debug using traditional debugging techniques. The problem is further compounded by the lack of debugging support on the devices. This is a serious problem because the device firmware is trusted by the operating system.Model checkers are designed to systematically verify properties of concurrent systems. Therefore, mod ...

13 A microarchitecture description language for retargeting firmware tools

J. F. Nixon, S. R. Schach, R. I. Winner

December 1986 **ACM SIGMICRO Newsletter , Proceedings of the 19th annual workshop on Microprogramming**, Volume 17 Issue 4

Full text available: pdf(1.02 MB) Additional Information: full citation, abstract, references, citings, index terms

ARCHI is a microarchitecture description language designed to serve as the underlying language for a retargetable firmware development environment. ARCHI provides a hierarchical, procedural description at the register transfer level of the target microarchitecture. This description can then be utilized by a meta-environment to generate an instantiation of the firmware development environment for a specific target microarchitecture. ARCHI has been successfully used to generate deb ...

14 Adaptation and personalization of VLSI-based computer architecture

Chiaki Ishikawa, Ken Sakamura, Mamoru Maekawa

December 1981 **ACM SIGMICRO Newsletter , Proceedings of the 14th annual workshop on Microprogramming**, Volume 12 Issue 4

Full text available: pdf(878.57 KB) Additional Information: full citation, abstract, references, index terms

This paper discusses the important problem of the adaptation and the personalization of VLSI-based computer systems by means of microprogramming. In order to fully exploit the VLSI chip with one million transistors on it, we propose the scheme of mass-producing the general purpose devices and then specializing them to particular application needs at the instruction set processor level and at upper levels of the computer systems. Many algorithms and examples are given. We show what facilitie ...

Keywords: Adaptation and personalization, Application-oriented architecture, Tuning of computer systems, VLSI-based microprogrammed computer architecture

15 A firmware monitor to support vertical migration decisions in the UNIX operating system

B. Holtkamp, H. Kaestner

October 1982 **ACM SIGMICRO Newsletter , Proceedings of the 15th annual workshop on Microprogramming**, Volume 13 Issue 4

Full text available: pdf(686.64 KB) Additional Information: full citation, abstract, references, citings, index terms

From a methodological point of view vertical migration involves the following four steps: identification of migration objects, prediction of expected system improvements, implementation, and verification of the results. In this paper a firmware monitor is presented as a support tool for the first and fourth step. The application environment for this monitor is a PDP-11/60 with writable control store running the UNIX operating system. Based upon a UNIX system model the requirement ...

16 Firmware factory & forth

Brad Eckert

December 1999 **ACM SIGPLAN Notices**, Volume 34 Issue 12

Full text available: pdf(373.72 KB) Additional Information: full citation, citings, index terms

17 An engineering environment for hardware/software co-simulation

D. Becker, R. K. Singh, S. G. Tell

July 1992 **Proceedings of the 29th ACM/IEEE conference on Design automation**

Full text available: pdf(583.00 KB) Additional Information: full citation, references, citings, index terms

18 VHDL and Ada firmware

D. L. Barton

July 1989 **Proceedings of the conference on TRI-Ada '88**

Full text available: pdf(414.23 KB) Additional Information: full citation, index terms

19 Heart: An operating system nucleus machine implemented by firmware

N. Kamibayashi, H. Ogawana, K. Nagayama, H. Aiso

March 1982 **Proceedings of the first international symposium on Architectural support for programming languages and operating systems**, Volume 17 , 10 Issue 4 , 2

Full text available: pdf(791.96 KB) Additional Information: full citation, abstract, references, index terms

This paper discusses the role of microprogramming in operating system design and shows several things: (1) advantages of the efficiency which may be gained from microcoded operating system primitives, (2) selecting the most appropriste primitives for implementation, and (3) an analysis of the tradeoffs among software, firmware, and hardware. The authors propose a practical approach of enhancing computer architecture level, from a view point of functional hierarchy of operating systems. In o ...

20 The implementation of a user-extensible system on a dynamically microprogrammable computer

Fergus K. Fung, Willis K. King

September 1977 **ACM SIGMICRO Newsletter , Proceedings of the 10th annual workshop on Microprogramming,** Volume 8 Issue 3

Full text available: pdf(308.59 KB) Additional Information: full citation, abstract, references, index terms

On a dynamically user-microprogrammable computer the user can tailor the machine to his needs by constructing microprogrammed routines and adding them to the system. If these routines are recognized by the assembler, then using them is no different from using any other basic machine instruction of the computer. The base machine is thus extended. The design and implementation of such a user-extensible system is described. It consists of 2 main parts: a pager which manages a virtual memory fo ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

PORTAL USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

virtual machine efi

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **virtual machine efi** Found **17,952** of **156,259**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Virtual memory mapped network interface for the SHRIMP multicomputer

Matthias A. Blumrich, Kai Li, Richard Alpert, Cezary Dubnicki, Edward W. Felten, Jonathan Sandberg

August 1998 **25 years of the international symposia on Computer architecture (selected papers)**

Full text available: pdf(1.39 MB) Additional Information: full citation, references, index terms

2 Parallel text retrieval on a high performance supercomputer using the Vector Space Model

P. Efraimidis, C. Glymidakis, B. Mamalis, P. Spirakis, B. Tampakas

July 1995 **Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval**

Full text available: pdf(959.41 KB) Additional Information: full citation, references, index terms

3 Using history of computing to address problems and opportunities

Orit Hazzan, John Impagliazzo, Raymond Lister, Shimon Schocken

February 2005 **ACM SIGCSE Bulletin , Proceedings of the 36th SIGCSE technical symposium on Computer science education**, Volume 37 Issue 1

Full text available: pdf(159.44 KB) Additional Information: full citation, references, index terms

Keywords: CS education research, computer science, computer science education, history of computing, software engineering

4 Technical reports

SIGACT News Staff

January 1980 **ACM SIGACT News**, Volume 12 Issue 1

Full text available: pdf(5.28 MB) Additional Information: full citation

5 The omni-directional treadmill: a locomotion device for virtual worlds
Rudolph P. Darken, William R. Cockayne, David Carmein
October 1997 **Proceedings of the 10th annual ACM symposium on User interface software and technology**
Full text available: pdf(1.30 MB) Additional Information: full citation, references, citings, index terms

Keywords: exertion devices, input devices, locomotion, maneuvering, virtual environments, virtual reality

6 An Abstract Machine for Symbolic Computation
D. L. Overheu
July 1966 **Journal of the ACM (JACM)**, Volume 13 Issue 3
Full text available: pdf(2.54 MB) Additional Information: full citation, abstract, references, index terms

The design of an abstract machine with a recursive function programming language which avoids the predicate type of conditional is described. It is shown that trough the adoption of list processing techniques it has been possible to construct a simple simulator for the machine in FORTRAN. A program for the machine which causes it to perform symbolic differentiation with some algebraic manipulation of the expressions concerned is given as an example of the type of computations which may be p ...

7 Intersecting solids on a massively parallel processor
Michael Karasick, David Strip
January 1995 **ACM Transactions on Graphics (TOG)**, Volume 14 Issue 1
Full text available: pdf(2.36 MB) Additional Information: full citation, abstract, references, index terms, review

Solid modeling underlies many technologies that are key to modern manufacturing. These range from CAD systems to robot simulators, from finite-element analysis to integrated circuit process modeling. The accuracy, and hence the utility, of these models is often constrained by the amount of computer time required to perform the desired operations. In this paper we present, in detail, an efficient algorithm for parallel intersections of solids using the Connection Machine, a massively paralle ...

Keywords: CAGD, grid data, hierarchical descriptions, spline surfaces, tensor product

8 Using lexicalized tags for machine translation
Anne Abeillé, Yves Schabes, Aravind K. Joshi
August 1990 **Proceedings of the 13th conference on Computational linguistics - Volume 3**
Full text available: pdf(569.06 KB) Additional Information: full citation, abstract, references, citings

Lexicalized Tree Adjoining Grammar (LTAG) is an attractive formalism for linguistic description mainly because of its extended domain of locality and its factoring recursion out from the domain of local dependencies (Joshi, 1985, Kroch and Joshi, 1985, Abeillé, 1988). LTAG's extended domain of locality enables one to localize syntactic dependencies (such as filler-gap), as well as semantic dependencies (such as predicate-arguments). The aim of this paper is to show that these properties c ...

9 Helper threads via virtual multithreading on an experimental itanium® 2 processor-based platform

Perry H. Wang, Jamison D. Collins, Hong Wang, Dongkeun Kim, Bill Greene, Kai-Ming Chan, Aamir B. Yunus, Terry Sych, Stephen F. Moore, John P. Shen

October 2004 **Proceedings of the 11th international conference on Architectural support for programming languages and operating systems**, Volume 39 , 38 , 32 Issue 11 , 5 , 5

Full text available: pdf(225.47 KB) Additional Information: full citation, abstract, references, index terms

Helper threading is a technology to accelerate a program by exploiting a processor's multithreading capability to run ``assist'' threads. Previous experiments on hyper-threaded processors have demonstrated significant speedups by using helper threads to prefetch hard-to-predict delinquent data accesses. In order to apply this technique to processors that do not have built-in hardware support for multithreading, we introduce virtual multithreading (VMT), a novel form of switch-on-event user-level ...

Keywords: DB2 database, PAL, cache miss prefetching, helper thread, itanium processor, multithreading, switch-on-event

10 Background execution of time warp programs

Christopher D. Carothers, Richard M. Fujimoto

July 1996 **ACM SIGSIM Simulation Digest , Proceedings of the tenth workshop on Parallel and distributed simulation**, Volume 26 Issue 1

Full text available: pdf(954.91 KB) Additional Information: full citation, references, citings, index terms

Publisher Site

11 A multimedia component kit: experiences with visual composition of applications

Vicki de Mey, Simon Gibbs

September 1993 **Proceedings of the first ACM international conference on Multimedia**

Full text available: pdf(87.59 KB) ps(303.61 KB) Additional Information: full citation, references, citings, index terms

Keywords: active objects, component-oriented software development, multimedia, visual composition

12 Inference of a 3-D object from a random partial 2-D projection

Pamela Hsu, Evangelos Triantaphyllou

April 1992 **Proceedings of the 1992 ACM/SIGAPP Symposium on Applied computing: technological challenges of the 1990's**

Full text available: pdf(729.10 KB) Additional Information: full citation, references, index terms

13 On the relation between functional and data parallel programming languages

Per Hammarlund, Björn Lisper

July 1993 **Proceedings of the conference on Functional programming languages and computer architecture**

Full text available: pdf(1.06 MB) Additional Information: full citation, references, citings, index terms

14 Fully-adaptive routing: packet switching performance and wormhole algorithms
S. A. Felperin, L. Gravano, G. D. Pifarré, J. L. C. Sanz
August 1991 **Proceedings of the 1991 ACM/IEEE conference on Supercomputing**
Full text available: pdf(978.41 KB) Additional Information: full citation, references, citings, index terms

15 Ada packages and distributed systems
Warren H. Jessop
February 1982 **ACM SIGPLAN Notices**, Volume 17 Issue 2
Full text available: pdf(830.33 KB) Additional Information: full citation, references, citings

16 Virtual machine monitors: Terra: a virtual machine-based platform for trusted computing
Tal Garfinkel, Ben Pfaff, Jim Chow, Mendel Rosenblum, Dan Boneh
October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**
Full text available: pdf(140.31 KB) Additional Information: full citation, abstract, references, citings, index terms

We present a flexible architecture for trusted computing, called Terra, that allows applications with a wide range of security requirements to run simultaneously on commodity hardware. Applications on Terra enjoy the semantics of running on a separate, dedicated, tamper-resistant hardware platform, while retaining the ability to run side-by-side with normal applications on a general-purpose computing platform. Terra achieves this synthesis by use of a *trusted virtual machine monitor* (TVMM ...

Keywords: VMM, attestation, authentication, trusted computing, virtual machine, virtual machine monitor

17 Virtual machine monitors: Xen and the art of virtualization
Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, Andrew Warfield
October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**
Full text available: pdf(168.76 KB) Additional Information: full citation, abstract, references, citings, index terms

Numerous systems have been designed which use virtualization to subdivide the ample resources of a modern computer. Some require specialized hardware, or cannot support commodity operating systems. Some target 100% binary compatibility at the expense of performance. Others sacrifice security or functionality for speed. Few offer resource isolation or performance guarantees; most provide only best-effort provisioning, risking denial of service.This paper presents Xen, an x86 virtual machine monit ...

Keywords: hypervisors, paravirtualization, virtual machine monitors

18 The design of a virtual machine for Ada

L. J. Groves, W. J. Rogers

November 1980 **ACM SIGPLAN Notices , Proceeding of the ACM-SIGPLAN symposium on Ada programming language**, Volume 15 Issue 11

Full text available: pdf(1.27 MB) Additional Information: full citation, abstract, references

An implementation of Ada should be based on a machine-independent translator generating code for a Virtual Machine, which can be realised on a variety of machines. This approach, which leads to a high degree of compiler portability, has been very successful in a number of recent language implementation projects and is the approach which has been specified by the U. S. Army and Air Force in their requirements for Ada implementations.This paper discusses the rationale, requirements and design of s ...

19 The case for virtual register machines

Brian Davis, Andrew Beatty, Kevin Casey, David Gregg, John Waldron

June 2003 **Proceedings of the 2003 workshop on Interpreters, virtual machines and emulators**

Full text available: pdf(238.65 KB) Additional Information: full citation, abstract, references, index terms

Virtual machines (VMs) are a popular target for language implementers. Conventional wisdom tells us that virtual stack architectures can be implemented with an interpreter more efficiently, since the location of operands is implicit in the stack pointer. In contrast, the operands of register machine instructions must be specified explicitly. In this paper, we present a working system for translating stack-based Java virtual machine (JVM) code to a simple register code. We describe the translatio ...

Keywords: interpreter, register architecture, stack architecture, virtual machine

20 Virtual machines: ReVirt: enabling intrusion analysis through virtual-machine logging and replay

George W. Dunlap, Samuel T. King, Sukru Cinar, Murtaza A. Basrai, Peter M. Chen

December 2002 **ACM SIGOPS Operating Systems Review**, Volume 36 Issue SI

Full text available: pdf(1.56 MB) Additional Information: full citation, abstract, references, citings

Current system loggers have two problems: they depend on the integrity of the operating system being logged, and they do not save sufficient information to replay and analyze attacks that include any non-deterministic events. ReVirt removes the dependency on the target operating system by moving it into a virtual machine and logging below the virtual machine. This allows ReVirt to replay the system's execution before, during, and after an intruder compromises the system, even if the intruder rep ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

PORTAL USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: The ACM Digital Library The Guide

virtual machine monitor itanium

Search

THE ACM DIGITAL LIBRARY

Feedback Report a problem Satisfaction survey

Terms used **virtual machine monitor itanium** Found **27,312** of **156,259**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Relevance scale

1 Helper threads via virtual multithreading on an experimental itanium® 2 processor-based platform

Perry H. Wang, Jamison D. Collins, Hong Wang, Dongkeun Kim, Bill Greene, Kai-Ming Chan, Aamir B. Yunus, Terry Sych, Stephen F. Moore, John P. Shen

October 2004 **Proceedings of the 11th international conference on Architectural support for programming languages and operating systems**, Volume 39 , 38 , 32 Issue 11 , 5 , 5

Full text available: pdf(225.47 KB) Additional Information: full citation, abstract, references, index terms

Helper threading is a technology to accelerate a program by exploiting a processor's multithreading capability to run ``assist'' threads. Previous experiments on hyper-threaded processors have demonstrated significant speedups by using helper threads to prefetch hard-to-predict delinquent data accesses. In order to apply this technique to processors that do not have built-in hardware support for multithreading, we introduce virtual multithreading (VMT), a novel form of switch-on-event user-level ...

Keywords: DB2 database, PAL, cache miss prefetching, helper thread, itanium processor, multithreading, switch-on-event

2 Prefetch injection based on hardware monitoring and object metadata

Ali-Reza Adl-Tabatabai, Richard L. Hudson, Mauricio J. Serrano, Sreenivas Subramoney

June 2004 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2004 conference on Programming language design and implementation**, Volume 39 Issue 6

Full text available: pdf(288.00 KB) Additional Information: full citation, abstract, references, index terms

Cache miss stalls hurt performance because of the large gap between memory and processor speeds - for example, the popular server benchmark SPEC JBB2000 spends 45% of its cycles stalled waiting for memory requests on the Itanium® 2 processor. Traversing linked data structures causes a large portion of these stalls. Prefetching for linked data structures remains a major challenge because serial data dependencies between elements in a linked data structure preclude the timely materialization ...

Keywords: cache misses, compiler optimization, garbage collection, prefetching,

profile-guided optimization, virtual machines

3 Lock reservation: Java locks can mostly do without atomic operations
Kiyokuni Kawachiya, Akira Koseki, Tamiya Onodera
November 2002 **ACM SIGPLAN Notices , Proceedings of the 17th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications**, Volume 37 Issue 11
Full text available: pdf(246.29 KB) Additional Information: full citation, abstract, references, citings, index terms

Because of the built-in support for multi-threaded programming, Java programs perform many lock operations. Although the overhead has been significantly reduced in the recent virtual machines, One or more atomic operations are required for acquiring and releasing an object's lock even in the fastest cases.This paper presents a novel algorithm called *lock reservation*. It exploits *thread locality* of Java locks, which claims that the locking sequence of a Java lock contains a very lon ...

Keywords: Java, atomic operation, lock, monitor, reservation, synchronization, thread locality

4 Optimizing indirect branch prediction accuracy in virtual machine interpreters
M. Anton Ertl, David Gregg
May 2003 **ACM SIGPLAN Notices , Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation**, Volume 38 Issue 5
Full text available: pdf(190.05 KB) Additional Information: full citation, abstract, references, index terms, review

Interpreters designed for efficiency execute a huge number of indirect branches and can spend more than half of the execution time in indirect branch mispredictions. Branch target buffers are the best widely available form of indirect branch prediction; however, their prediction accuracy for existing interpreters is only 2%--50%. In this paper we investigate two methods for improving the prediction accuracy of BTBs for interpreters: replicating virtual machine (VM) instructions and combining seq ...

Keywords: branch prediction, branch target buffer, code replication, interpreter, superinstruction

5 Pinpointing Representative Portions of Large Intel® Itanium® Programs with Dynamic Instrumentation
Harish Patil, Robert Cohn, Mark Charney, Rajiv Kapoor, Andrew Sun, Anand Karunanidhi
December 2004 **Proceedings of the 37th annual International Symposium on Microarchitecture**
Full text available: pdf(224.59 KB) Additional Information: full citation, abstract

Detailed modeling of the performance of commercial applications is difficult. The applications can take a very long time to run on real hardware and it is impractical to simulate them to completion on performance models. Furthermore, these applications have complex execution environments that cannot easily be reproduced on a simulator, making porting the applications to simulators difficult. We attack these problems using the well-known SimPoint methodology to find representative portions of an ...

6 Field-testing IMPACT EPIC research results in Itanium 2

John W. Sias, Sain-zee Ueng, Geoff A. Kent, Ian M. Steiner, Erik M. Nystrom, Wen-mei W. Hwu

March 2004 **ACM SIGARCH Computer Architecture News , Proceedings of the 31st annual international symposium on Computer architecture - Volume 00**, Volume 32 Issue 2

Full text available: pdf(273.71 KB) Additional Information: full citation, abstract

Explicitly-Parallel Instruction Computing (EPIC) providesarchitectural features, including predication and explicitcontrol speculation, intended to enhance the compiler'sability to expose instruction-level parallelism (ILP) incontrol-intensive programs. Aggressive structural transformationsusing these features, though described in theliterature, have not yet been fully characterized in completesystems. Using the Intel Itanium 2 microprocessor,the SPECint2000 benchmarks and the IMPACT Compilerfor ...

7 EPIC compilation: Optimizations to prevent cache penalties for the Intel® Itanium® 2 Processor

Jean-Francois Collard, Daniel Lavery

March 2003 **Proceedings of the international symposium on Code generation and optimization: feedback-directed and runtime optimization**

Full text available: pdf(1.28 MB) Publisher Site Additional Information: full citation, abstract, references, index terms

This paper describes scheduling optimizations in the Intel® Itanium® compiler to prevent cache penalties due to various micro-architectural effects on the Itanium 2 processor. This paper does not try to improve cache hit rates but to avoid penalties, which probably all processors have in one form or another, even in the case of cache hits. These optimizations make use of sophisticated methods for disambiguation of memory references, and this paper examines the performance improvement obt ...

8 The Performance of Runtime Data Cache Prefetching in a Dynamic Optimization System

Jiwei Lu, Howard Chen, Rao Fu, Wei-Chung Hsu, Bobbie Othmer, Pen-Chung Yew, Dong-Yuan Chen

December 2003 **Proceedings of the 36th annual IEEE/ACM International Symposium on Microarchitecture**

Full text available: pdf(253.79 KB) Additional Information: full citation, abstract, citings, index terms

Traditional software controlled data cache prefetching isoften ineffective due to the lack of runtime cache miss andmiss address information. To overcome this limitation, weimplement runtime data cache prefetching in the dynamicoptimization system ADORE (ADaptive Object code RE-optimization).Its performance has been compared withstatic software prefetching on the SPEC2000 benchmarksuite. Runtime cache prefetching shows better performance.On an Itanium 2 based Linux workstation, it can increasepe ...

9 ULT: a Java threads model for platform independent execution

Ruben Pinilla, Marisa Gil

October 2003 **ACM SIGOPS Operating Systems Review**, Volume 37 Issue 4

Full text available: pdf(3.39 MB) Additional Information: full citation, abstract, references

Java is known to be a valuable technology for building platform independent applications, based on an independent execution environment provided by a virtual machine (JVM, Java Virtual Machine) and an API formed by a set of classes. The Java platform was conceived as a solution for application transportation between heterogeneous platforms without the need of adapting and recompiling the source

code. Some previous analyses of Sun JVM implementation (Java 2 SDK 1.2.2-006) establish that the HPI (...

Keywords: HPI, JVM, Java threads, ULT, concurrency, kernel threads, multithreaded, scheduling, user threads

10 Compilation: The impact of if-conversion and branch prediction on program execution on the Intel® Itanium™ processor

Youngsoo Choi, Allan Knies, Luke Gerke, Tin-Fook Ngai

December 2001 **Proceedings of the 34th annual ACM/IEEE international symposium on Microarchitecture**

Full text available: pdf(1.06 MB) Publisher Site Additional Information: full citation, abstract, references, citings

The research community has studied if-conversion for many years. However, due to the lack of existing hardware, studies were conducted by simulating code generated by experimental compilers. This paper presents the first comprehensive study of the use of predication to implement if-conversion on production hardware with a near-production compiler. To better understand trends in the measurements, we generated binaries at three increasing levels of if-conversion aggressiveness. For each level, we ...

11 A proposal for a new hardware cache monitoring architecture

Martin Schulz, Jie Tao, Jürgen Jeitner, Wolfgang Karl

June 2002 **ACM SIGPLAN Notices , Proceedings of the workshop on Memory system performance**, Volume 38 Issue 2 supplement

Full text available: pdf(1.23 MB) Additional Information: full citation, abstract, references

The analysis of the memory access behavior of applications, an essential step for a successful cache optimization, is a complex task. It needs to be supported with appropriate tools and monitoring facilities. Currently, however, users can only rely on either simulation based approaches, which deliver a large degree of detail but are restricted in their applicability, or on hardware counters embedded into processors, which allow to keep track of very few, mostly global events and hence only provi ...

12 Virtual machine monitors: Terra: a virtual machine-based platform for trusted computing

Tal Garfinkel, Ben Pfaff, Jim Chow, Mendel Rosenblum, Dan Boneh

October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**

Full text available: pdf(140.31 KB) Additional Information: full citation, abstract, references, citings, index terms

We present a flexible architecture for trusted computing, called Terra, that allows applications with a wide range of security requirements to run simultaneously on commodity hardware. Applications on Terra enjoy the semantics of running on a separate, dedicated, tamper-resistant hardware platform, while retaining the ability to run side-by-side with normal applications on a general-purpose computing platform. Terra achieves this synthesis by use of a *trusted virtual machine monitor* (TVMM ...

Keywords: VMM, attestation, authentication, trusted computing, virtual machine, virtual machine monitor

13 Frontmatter (includes Activities, Conference Corner, and Technical Correspondence)
SIGPLAN Notices staff
August 2003 **ACM SIGPLAN Notices**, Volume 38 Issue 8
Full text available: pdf(629.55 KB) Additional Information: full citation

14 Bytecode fetch optimization for a Java interpreter
Kazunori Ogata, Hideaki Komatsu, Toshio Nakatani
October 2002 **Proceedings of the 10th international conference on Architectural support for programming languages and operating systems**, Volume 36 , 37 , 30 Issue 5 , 10 , 5
Full text available: pdf(1.16 MB) Additional Information: full citation, abstract, references, citings, index terms

Interpreters play an important role in many languages, and their performance is critical particularly for the popular language Java. The performance of the interpreter is important even for high-performance virtual machines that employ just-in-time compiler technology, because there are advantages in delaying the start of compilation and in reducing the number of the target methods to be compiled. Many techniques have been proposed to improve the performance of various interpreters, but none of ...

Keywords: Java, PowerPC, bytecode interpreter, performance, pipelined interpreter, stack caching, superscalar processor

15 An implementation scheme for a virtual machine monitor to be realized on user - microprogrammable minicomputers
B. D. Shriver, J. W. Anderson, L. J. Waguespack, D. M. Hyams, R. A. Bombet
October 1976 **Proceedings of the annual conference**
Full text available: pdf(654.60 KB) Additional Information: full citation, abstract, references, citings, index terms

A virtual machine monitor allows several different operating systems to run concurrently on the same machine. This paper presents the description of a virtual machine monitor and its support structure which can be implemented on a microprogrammable minicomputer or a distributed network of such machines. In our approach, all storage, transformational, input, and output resources of the system are accessed through a mapping mechanism. The design and implementation methodology for an actual re ...

16 The Inca Test Harness and Reporting Framework
Shava Smallen, Catherine Olschanowsky, Kate Ericson, Pete Beckman, Jennifer M. Schopf
November 2004 **Proceedings of the 2004 ACM/IEEE conference on Supercomputing**
Full text available: pdf(3.01 MB) Additional Information: full citation, abstract

Virtual organizations (VOs), communities that enable coordinated resource sharing among multiple sites, are becoming more prevalent in the high-performance computing community. In order to promote cross-site resource usability, most VOs prepare service agreements that include a minimum set of common resource functionality, starting with a common software stack and evolving into more complicated service and interoperability agreements. VO service agreements are often difficult to verify and maint ...

17 OS Debugging Method Using a Lightweight Virtual Machine Monitor
Tadashi Takeuchi
March 2005 **Proceedings of the conference on Design, Automation and Test in Europe - Volume 2**
Full text available: pdf(85.92 KB) Additional Information: full citation, abstract

Demands for implementing original OSs that can achieve high I/O performance on PC/AT compatible hardware have recently been increasing, but conventional OS debugging environments have not been able to simultaneously assure their stability, be easily customized to new OSs and new I/O devices, and assure efficient execution of I/O operations. We therefore developed a novel OS debugging method using a lightweight virtual machine. We evaluated this debugging method experimentally and confirmed that ...

18 Virtual machine monitors: Xen and the art of virtualization
Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, Tim Harris, Alex Ho, Rolf Neugebauer, Ian Pratt, Andrew Warfield
October 2003 **Proceedings of the nineteenth ACM symposium on Operating systems principles**
Full text available: pdf(168.76 KB) Additional Information: full citation, abstract, references, citings, index terms

Numerous systems have been designed which use virtualization to subdivide the ample resources of a modern computer. Some require specialized hardware, or cannot support commodity operating systems. Some target 100% binary compatibility at the expense of performance. Others sacrifice security or functionality for speed. Few offer resource isolation or performance guarantees; most provide only best-effort provisioning, risking denial of service.This paper presents Xen, an x86 virtual machine monit ...

Keywords: hypervisors, paravirtualization, virtual machine monitors

19 Track 7: compilers and operating systems: Dynamic run-time architecture techniques for enabling continuous optimization
Tipp Moseley, Alex Shye, Vijay Janapa Reddi, Matthew Iyer, Dan Fay, David Hodgdon, Joshua L. Kihm, Alex Settle, Dirk Grunwald, Daniel A. Connors
May 2005 **Proceedings of the 2nd conference on Computing frontiers**
Full text available: pdf(347.97 KB) Additional Information: full citation, abstract, references, index terms

Future computer systems will integrate tens of multithreaded processor cores on a single chip die, resulting in hundreds of concurrent program threads sharing system resources. These designs will be the cornerstone of improving throughput in high-performance computing and server environments. However, to date, appropriate systems software (operating system, run-time system, and compiler) technologies for these emerging machines have not been adequately explored. Future processors will require so ...

Keywords: multithreading, performance counters, profiling, scheduling

20 Compilers II: Inter-procedural stacked register allocation for itanium® like architecture
Liu Yang, Sun Chan, G. R. Gao, Roy Ju, Guei-Yuan Lueh, Zhaoqing Zhang
June 2003 **Proceedings of the 17th annual international conference on Supercomputing**
Full text available: pdf(478.20 KB) Additional Information: full citation, abstract, references, index terms

A hardware managed register stack, Register Stack Engine (RSE), is implemented in Itanium® architecture to provide a unified and flexible register structure to software. The compiler allocates each procedure a register stack frame with its size explicitly specified using an *alloc* instruction. When the total number of registers used by the procedures on the call stack exceeds the number of physical registers, RSE performs automatically register overflows and fills to ensure that the c ...

Keywords: hot region, hotspot, inter-procedural stacked register allocation, quota assignment, register allocation

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.
Terms of Usage Privacy Policy Code of Ethics Contact Us

Useful downloads: Adobe Acrobat QuickTime Windows Media Player Real Player

Extensible Firmware Interface

EFI Presentations

All slide shows are in Adobe* Acrobat* format.
Download Adobe Acrobat Reader

Fall 2004 IDF

- EFI and Windows "Longhorn" – Microsoft Corporation
  WMV Downloads: part 1 part 2 part 3 part 4
- Next Generation EFI Operating System "Linux & EFI" - Intel Corporation
  WMV Downloads: part 1 part 2 part 3 part 4
- Framework Fall 2004 IDF Presentations

Spring 2004 IDF

- Cross Platform Management and Provisioning with the Intel Platform Innovation Framework for EFI - Intel Corporation
  WMV Downloads: part 1 part 2 part 3
- Windows*, EFI and Testing (Supporting 32 and 64-bit platforms) - Microsoft Corporation
  WMV Downloads: part 1 part 2

Relevant presentations from Microsoft on EFI

- Winhec*
  - EFI and Windows Longhorn [WinHEC 2004] - EFI plans for Windows Longhorn
  - Implementing EFI on 32-bit Systems [WinHEC 2004] - Implementation plans for EFI on 32-bit computers
- EFI-UGA Binding*
- EFI 1.x Solutions with AMIBIOS8*

Fall 2003 IDF

- Windows, EFI and Testing - Intel Corporation and Microsoft Corporation
- Writing and Debugging EFI Drivers - Intel Corporation and Hewlett Packard Co.
- EFI Specification Evolution

Spring 2003 IDF

- EFI 1.10 and Beyond: An Overview
- Manufacturing and Test solutions with EFI
- EFI Driver's Writers Guide - Intel Corporation and Hewlett Packard Co.
- From Research and Development at Intel: Beyond the BIOS
- Evolving Firmware for Remote Manageability

Fall 2002 IDF

- EFI 1.10 and Beyond: An Overview
- Evolving Firmware for Remote Manageability
- Developing a Production EFI Driver - Adaptec Corp.
- Case Study: Enhancing Pre-boot Environment with EFI Applications - Inside Corp.
- Replacing VGA: Theory, Practice and Demo of EFI and UGA - ATI Technologies Inc. (ATI), Microsoft Corporation, Intel Corporation
- Hands-On EFI Debugging - American Megatrends, Inc. (AMI)

Spring 2002 IDF

- Extensible Firmware Interface - Changing the Face of BIOS
- EFI Byte Code Driver Development
- EFI & UGA: Pre-boot & Runtime Environment - ATI Technologies Inc. (ATI), Microsoft Corporation, Intel Corporation
- EFI Platform Support Tools: Debug/Diagnose/Recover - American Megatrends, Inc. (AMI)
- Linux Support for EFI on Itanium™ Processor Family & IA-32 Platforms - Hewlett Packard Co.
- Windows CE* and EFI - Loronix Information Systems

LABS

- Using EFI to Manage Pre-boot Environment
- EFI Driver Lab

Fall 2001 IDF

- Extensible Firmware Interface - Changing the Face of BIOS
- EFI 1.1 Driver Model and Protocol Services Overview
- EFI 1.1 Driver Model Demonstration
- Case Study: Porting to the EFI 1.1 Driver Model - LSI Logic Corporation
- Extending EFI the Right Way: Case Studies - Hewlett Packard Co.
- Building Extensions for the EFI Shell
- Porting DOS*-Based RDRAM* Channel Characterization Tools to EFI: A Case Study - Rambus Inc.
- EFI 1.1 Universal Graphics Adapter (UGA)
- Debugging C-Source Code for EFI - American Arium

Spring 2001 IDF

- EFI Overview
- EFI 1.1 - Driver Writer's Guide
- Writing EFI Manufacturing Tests - American Megatrends, Inc. (AMI)
- ACPI Protocol Driver for EFI
- Moving DOS Tools to EFI - A Case Study - PowerQuest Corporation

October 2000 - Training

- EFI Training Session

Fall 2000 IDF

- Implementing EFI 1.0
- EFI 1.1 preview - replacement of legacy Option ROMs, EFI Driver Model, and EFI Virtual Machine
- PXE 32/64 - integration of network boot capability in EFI
- Microsoft* Windows* and EFI, EFI Disk Utilities, and Introduction to UGA - Universal Graphics Adapter - Microsoft Corporation

Spring 2000 IDF

- EFI and the EFI Application Toolkit

EFI legal information

back to top

Site Index | *Legal Information | Privacy Policy | Contact Us

©2005 Intel Corporation